K:\JSMITH\OSA\Osa96-01.REP
JMS75
JMS/cjh
06/22/99

PATENT APPLICATION
Attorney's Docket No.: OSA96-01

#13
KW
6-25-99

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicants: Clifford Heath, Graeme Port, Steven Klos and Graeme Greenhill

Application No.: 08/707,622 Group: ~~2764~~ 2767

Filed: September 5, 1996 Examiner: M. Smithers

For: Systems and Methods for Automatic Application Version Upgrading and Maintenance

CERTIFICATE OF FACSIMILE TRANSMISSION
I hereby certify that this correspondence is being facsimile transmitted to the United States Patent and Trademark Office:

on 6-22-99 (Date) (Signature)

CLAIRE T. HANDLEY
Typed or printed name of person signing certificate

REPLY

Assistant Commissioner for Patents
Washington, D.C. 20231

Sir:

This reply is being filed in response to the Office Action mailed from the U.S. Patent and Trademark Office on February 16, 1999 in the above-identified application.

A two-month extension of time to respond to the Office Action is respectfully requested. A Petition for a two-month extension of time is being filed concurrently together with the appropriate fee.

Applicant thanks Examiner Smithers for a helpful telephone interview with the undersigned. A draft response was considered and has been revised as presented here to address the Examiner's comments by including further discussion of the Brichta et al. reference and of the claims.

It is hoped that this response will place the application in condition for allowance. If not, in view of the special status of the application, a telephone call from the Examiner to address any outstanding issues would be appreciated.

In the Office Action dated February 16, 1999, all claims were rejected under 35 U.S.C. § 103 as being unpatentable over U.S. Patent to Kullick et al. and Brichta et al., alone or in combination with Cole et al., Arnold et al., Guarneri et al., Moore, Butts et al., Kikinis and/or de Hond. The rejections are respectfully traversed and reconsideration is requested.

The invention is directed to a method and system for updating application program components on a client through a network. In the preferred embodiment, the client includes a program which performs all update processing, the server system only being required to download files. Specifically, a client which is to update the application program components first downloads a catalog file from the server. The catalog file identifies the most recent versions of the program components as well as the network addresses where those components are stored. The program in the client then compares the version identifications in the catalog file with versions then on the client and, where a new version is required, downloads the new version from the network address indicated in the catalog file. In the preferred implementation, the updating program is a launcher program which serves as a proxy to update the application program when a user selects the application program for execution.

Since all processing is performed at the client, the server need only make available the catalog files and application program components for standard file transfer. No proprietary software is required on the server, and the required catalog and component files can be stored on any file server accessible to the client. Further, the only communications required between the client and the server are to first download the catalog file and to subsequently download any required components.

Of the references cited, only Kullick et al. and Cole et al. relate to the update of software programs on a client. Cole et al. was discussed in detail and distinguished in the last response.

In Kullick et al., the management program on the client "checks the shared memory area 14 to determine whether an upgrade version is present, and if so whether that version is more recent than the newest version of the application stored in the local memory 16. If so, the management program downloads a copy of the most recent version to the memory 16 of the client computer . . . . Of particular significance, control of the updating procedure takes place from within the client computer itself, so that no external mechanisms are required to implement

the automatic updating function." (Column 4, lines 6-19.) Kullick et al. does not indicate how the client determines whether an upgrade version is present. However, as noted by the Examiner, Kullick et al. fails to disclose a catalog of components with the version identification. Kullick et al. is apparently unable to distinguish between entire program versions and versions of components of programs. Accordingly, it would be necessary to download an entire program even if the only changes from an earlier version were in a small component of the overall program.

In accordance with the present invention, the updating process is carried out only after the client has downloaded a catalog for the identified application program. The client compares the version identifications of the components maintained on the server, indicated in the catalog, and of the components maintained on the client. Then, only the selected components need by downloaded by the client.

Brichta et al. has been cited for the showing of "the use of a catalog data base upgrade system which is capable of adding and modifying items in the data base." However, Brichta et al. has nothing to do with updating programs stored on a client in a network. Rather, Brichta et al. is concerned with catalogs in general, such as inventory catalogs, and with updating the catalogs themselves. There is no suggestion in either Kullick et al. or Brichta et al. that a catalog, and in particular a program component catalog, would be useful in providing a more efficient program updating process.

Examiner Smithers specifically questioned the Brichta et al. statement in the sentence bridging columns 3 and 4 that "catalog upgrade system 34 may include one or more computer programs." The programs referenced in that sentence are the programs which operate on the catalog to create and upgrade the catalog. It is not a statement that the catalog lists computer programs or, more particularly, versions of component program as in the present invention. More specifically, it is further stated at the top of column 4 that the catalog upgrade system 34 performs the functions described with reference to Figures 2, 3 and 4 of Brichta et al. Figure 2 is a create/upgrade routine for creating and modifying the catalog. Figure 3 is a flow chart for adding and modifying catalog entries, and Figure 4 is a routine for navigating a catalog. These are generic processes for handling a catalog without regard for the nature of the items listed in the catalog. There is no suggestion that those items be program component versions or that the catalog be used in a method for updating programs on the client. In fact, specific items envisioned by Brichta et al. are exemplified by the items in Figures 5, 6 and 7.

Examiner Smithers also questioned why the catalog of Brichta et al. could not be used to list program components. It is not known whether the catalog system of Brichta et al. would be suitable for use in the claimed invention. However, Brichta et al. does not teach such use. Any invention makes use of known components. The question is not whether those components can be used in the invention, but whether the prior art teaches the use of the components in the combination claimed. Such teachings cannot be found in the cited references.

More specifically, claim 1 recites maintaining a catalog of components of the program with version identification. Such a catalog is not suggested in any of the prior art references. Further, claim 1 recites that the server downloads to the client the catalog for the identified application program in response to a call from the client and the client compares the version identification between the components indicated in the catalog and the components maintained in the client. Again, no such steps are suggested in the cited references.

Similarly, claim 31 recites "a catalog on the server for specifying the components with the version identification." Claim 31 further recites that a client "causes the server to download the catalog to the client, compares the version identifications of the components maintained on the server, indicated in the downloaded catalog, and the version identifications of the components maintained on the client." Such identification of version components is not suggested in the prior art references.

Similarly, claim 60 recites a launcher program which "causes a server to download a catalog to the client, the catalog specifying the components with the version identification." Further, claim 60 recites that the launcher compares the version identification of the components indicated in the downloaded catalog with version identifications of the components maintained on the client and updates the application program. There is no suggestion of those steps of updating the application program in the cited references.

Claim 61 recites instructions for execution on a client to cause a server to download a catalog of components with the version identification. The claim further recites that the version identifications indicated in the catalog are compared with the version identifications of the components maintained on the client. Those features are not suggested by the prior art.

Claim 62 recites creating a catalog file which lists the names of all the required components of each of the application programs and specifying for each component a current version identification and either a content of the component or a network address from which the component can be retrieved. The claim further recites a launcher program which retrieves the

current version of the catalog file and compares the components and their version identification to corresponding information in another catalog of application components already stored on the client computer. The claim further recites retrieving any components which the launcher program determines as either not present or having incorrect version identification. Those features are not suggested in the prior art.

Claim 65 recites maintaining a catalog of components with version identifications and causing the server to download the catalog to the client in response to a call from the client. The claim further recites, in response to a client call from the client, causing the server to download a second catalog including the latest version identifications of the components and any new additional components on the server. Further, claim 65 recites comparing the latest version identifications of the components in the second catalog with the version identifications of the components maintained on the client from the first catalog. These features are not suggested in the prior art.

The remaining independent claims are based on the original claim 1. Each recites maintaining a catalog of components of a program with version identification. Each also recites a client causing the server to download the catalog to the client and the client comparing the version identification between the components maintained on the server, indicated in the catalog, and the components maintained on the client. Those features are not suggested in the prior art references.

The various dependent claims in the present application are directed to specific features which are advantageously used in the updating of programming components using a component catalog downloaded to a client. Since none of the references teach that aspect of the invention, none of the cited references, which are generally unrelated to either program updating or to catalogs, can be said to teach the features of the dependent claims.

CONCLUSION

In view of the above remarks, it is believed that all claims are in condition for allowance, and it is respectfully requested that the application be passed to issue. If the Examiner feels that a telephone conference would expedite prosecution of this case, the Examiner is invited to call the undersigned at (781) 861-6240.

Respectfully submitted,

HAMILTON, BROOK, SMITH & REYNOLDS, P.C.

By _____

James M. Smith, Esq.
Registration No. 28,043
Telephone (781) 861-6240
Facsimile (781) 861-9540

Lexington, Massachusetts 02421-4799

Dated: 6/22/99